ACADEMIC PRESS

Available online at www.sciencedirect.com
SCIENCE DIRECT®

Journal of Computational Physics 186 (2003) 1–23

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

A least square extrapolation method for improving solution accuracy of PDE computations

M. Garbey [a], W. Shyy [b,*]

[a] Department of Computer Science, University of Houston, Houston, TX 77204, USA
[b] Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611, USA

Received 10 January 2002; received in revised form 3 December 2002; accepted 16 December 2002

Abstract

Richardson extrapolation (RE) is based on a very simple and elegant mathematical idea that has been successful in several areas of numerical analysis such as quadrature or time integration of ODEs. In theory, RE can be used also on PDE approximations when the convergence order of a discrete solution is clearly known. But in practice, the order of a numerical method often depends on space location and is not accurately satisfied on different levels of grids used in the extrapolation formula. We propose in this paper a more robust and numerically efficient method based on the idea of finding automatically the order of a method as the solution of a least square minimization problem on the residual. We introduce a two-level and three-level least square extrapolation method that works on nonmatching embedded grid solutions via spline interpolation. Our least square extrapolation method is a post-processing of data produced by existing PDE codes, that is easy to implement and can be a better tool than RE for code verification. It can be also used to make a cascade of computation more numerically efficient. We can establish a consistent linear combination of coarser grid solutions to produce a better approximation of the PDE solution at a much lower cost than direct computation on a finer grid. To illustrate the performance of the method, examples including two-dimensional turning point problem with sharp transition layer and the Navier–Stokes flow inside a lid-driven cavity are adopted.

1. Introduction

Richardson extrapolation (RE) is based on a very simple and elegant mathematical idea. It has been used successfully in several areas of numerical analysis such as quadrature with the Romberg method or ODE integrations that have smooth enough solution with the Bulirsch-Stoer method [21]. Its use in practical situation such as computational fluid dynamics (CFD) is however questionable, because it is rare that all mathematical hypothesis needed by RE are fulfilled by the numerical approximation. In most cases the convergence estimates required by RE are not rigorously justified for the full CFD model. They are often

* Corresponding author.
E-mail address: wss@mae.ufl.edu (W. Shyy).

doi:10.1016/S0021-9991(02)00080-3

deduced from simplified model equations. Furthermore, in three-dimension (3D), the meshes are usually not fine enough to satisfy accurately the a priori convergence estimates that are only asymptotic relations in nature. Finally the order of convergence of a CFD code is generally space dependent and eventually solution dependent. One requires then three grid levels to provide an estimation of the order of convergence. It is unlikely in 3D computation of complex flow problems that one can provide two or three solutions, on progressively refined grids and with a coarse grid solution that has a satisfactory level of accuracy, to be used in RE.

We propose a general extrapolation method, based on a least square approximation of order of convergence, and will present both theoretical analysis and computational demonstrations with steady state problems. We symbolically formulate our least square problem as follows: Let u_i, $i = 1..3$ be a family of approximate solutions of the PDE problem $N[u] = 0$, provided on a family of meshes M_i. In the two-level case, we construct the space dependant function α that minimizes

$$N[\alpha u_1 + (1 - \alpha)u_2], \tag{1}$$

in some L_2 space on a grid much finer than M_i, $i = 1..3$.

In the three-level case, we construct a pair of space functions (α, β) that minimizes

$$N[\alpha u_1 + \beta u_2 + (1 - \alpha - \beta)u_3], \tag{2}$$

in some L_2 space on a grid much finer than M_i, $i = 1..3$. This is a natural generalization of RE, since in RE, one restricts the weight functions α and β to be space independent constants.

In this paper, we consider a situation where typically the u_i functions are finite differences (FD), or finite volume (FV) approximations of the Navier–Stokes equations. We then look for spectral approximation of the weight functions with few modes. We will show in practical situations that

- Provided that one can obtain the definition of the residual of the PDE approximation, the existence of a stability estimate on the approximation of the PDE's problem and two grid solutions with increasing accuracy, the method can find automatically the order of convergence.
- Using three different grid solutions (not necessarily with uniformly increasing mesh resolution) our method provides a solution with improved accuracy.

It is also observed that our extrapolation formula re-scales the problem since the weight function should be of order one. Furthermore, the order of convergence of the fine grid solution produced is based on the multiplication of the coarse grid solution accuracy times the weight functions accuracy. The accuracy requirement on α and β functions is therefore very modest and consequently the computational cost of the least square problem itself will be negligible compare to a direct computation of the fine grid solution. In this paper we have restricted ourselves to the linear least square theory, and tackle a nonlinear problem via a Newton-like loop that starts from one of the grid solution u_i. Nevertheless, our least square extrapolation combined with a multilevel approach is general, and can be applied to variational formulation problems (for example with Finite Element) as well as FV formulation. In this paper, we use mainly a modified Fourier expansion for the unknown extrapolation weight functions. We will also present an overlapping domain decomposition version of our least square problem that is useful to introduce some adaptivity. Furthermore, since there are no boundary conditions required on α and β, complex geometry should be naturally handled by fictitious domains with appropriate extension of u_i in regular shaped domains.

A major goal of our method is to have a practical and simple tool to enhance CFD accuracy and efficiency in the context of code verification.

In this paper, we illustrate the potential of the method with singular perturbation problem as a two-dimensional (2D) turning point problem with sharp internal transition layer. Further, we show some promising results for incompressible flows such as the cavity flow problem. We demonstrate first with the incompressible Navier–Stokes flow inside a lid-driven cavity that the RE method is not adequate. Then we demonstrate that, as opposed to RE, we can gain one order of accuracy with least square extrapolation provided that the coarse grid solution meets certain accuracy requirement. To be more practical, we use a cascade of *nonmatching embedded* grid solution process. Such a practice is particularly useful in 3D computation because in most cases one cannot afford three levels of uniform refinement while the coarse grid solution provides still reasonable accuracy. In all these examples, the computational cost of our method is negligible compared to the fine grid solution. Finally we believe that our least square extrapolation method should be a more efficient tool than RE in the code verification context [3,13,15,17,19]. In particular, the present method can be combined to cascade algorithm as discussed in [4–6,20].

The plan of the paper is as follows. In Section 2, we summarize basic properties of RE method and evaluate its application to CFD. In Section 3, we present some basic approximation theory for least square extrapolation applied to grid functions. In Section 4, we extend this technique to PDEs, and report on numerical results of our method for turning point problem and steady incompressible Navier–Stokes flows.

2. Basic properties of RE and computational implications

Let us first summarize some basic properties of RE in the context of approximation functions.

2.1. Extrapolation in a normed linear space

We are going to review briefly the properties of Richardson extrapolation. We will present two complementary points of view that are asymptotic expansion for continuous function in a normed vector space, and numerical approximation for discrete functions defined on a mesh. We will use lower case u for continuous function and upper case U for numerical approximation.

Let E be a normed linear space, and $\| \ \|$ its norm. Let p be a positive integer, h_0 a positive real and $h \in (0, h_0)$. Let v be an element of E, and $u^i \in E$, $i = 1..3$ that have the following asymptotic expansion [1],

$$u^i = v + C\left(\frac{h}{2^{i-1}}\right)^p + \delta, \tag{3}$$

with C positive constant independent of h, and $\|\delta\| = \mathrm{o}(h^p)$. If p is known then v_r^i defined by the RE formula,

$$v_r^i = \frac{2^p u^{i+1} - u^i}{2^p - 1}, \quad i = 1, 2 \tag{4}$$

satisfies

$$\|v - v_r^i\| = \mathrm{o}(h^p)$$

and is therefore an improved approximation of v compared to each of the u^i approximations.

From the numerical point of view, let E_h be a family of normed linear space $(E_h, \| \ \|)$, associated with a mesh M_h used to approximate elements of $(E, \| \ \|)$ of p^{th} order. For simplicity, we restrict ourselves to three embedded meshes $M_{h/4} \subset M_{h/2} \subset M_h$. We denote then E_i the linear space corresponding to mesh $M_{h/2^{i-1}}$ with $E_1 \subset E_2 \subset E_3 \subset E$.

One rewrites (3) in E_i, as a set of equations,

$$U^i = v + C_i\left(\frac{h}{2^{i-1}}\right)^p + \delta_i \tag{5}$$

with $C_i = (1 + \epsilon_i)C$, and $\epsilon_i = o(1)$. In these equations $C(h/(2^{i-1}))^p$ is an approximation of the error of first order. ϵ_i is for the second-order error term. δ_i is a model for the h independent numerical perturbation induced by consistency errors and/or arithmetic error. For example, in CFD calculation, one can use an iterative solver to reach the discrete functions U_i, $i = 1..3$ and δ_i results from imperfect convergence of the iterative solver. The Richardson extrapolate

$$V_r^2 = \frac{2^p U^3 - U^2}{2^p - 1}, \tag{6}$$

has then for error in E_1,

$$v - V_r^2 = \frac{1}{2^p - 1}\left((\delta_2 - 2^p \delta_3) + C(\epsilon_2 - \epsilon_3)\left(\frac{h}{2}\right)^p\right). \tag{7}$$

The numerical perturbation is therefore weakly amplified by a factor $(2^p + 1)/(2^p - 1)$.

Back to the asymptotic point of view, if the asymptotic expansion of u^i is known at next order as for example

$$u^i = v + C\left(\frac{h}{2^{i-1}}\right)^p + \tilde{C}\left(\frac{h}{2^{i-1}}\right)^q + \delta, \tag{8}$$

with q positive integer larger than p and $\|\delta\| = o(h^q)$, one can in principle eliminate h^p and h^q terms and get a better estimate by combining u^1, u^2, u^3 as

$$v_r^i = \frac{u^1 - (2^p + 2^q)u^2 + 2^{p+q}u^3}{(1 - 2^p)(1 - 2^q)}. \tag{9}$$

But for typical applications in CFD calculation the asymptotic order of convergence is not well established [12,17–19,26,27].

One can derive from the set of all three asymptotic relations (3), the asymptotic estimate

$$p \sim \log_2 \frac{\|u^1 - u^2\|}{\|u^2 - u^3\|}. \tag{10}$$

If one considers $\{u^i\}$ as a set of *real numbers* instead of a set of functions in $(E, \| \|)$, combining (4) and (10) gives the so-called Δ^2 Aitken formula [11],

$$v_r^2 \sim \frac{u^1 u^3 - (u^2)^2}{u^1 - 2u^2 + u^3}. \tag{11}$$

One can use (11) in a pointwise manner in CFD. However this formula has generally no rigorous basis in the corresponding space of approximation.

From the numerical point of view, one gets from (5),

$$p = \log_2 \left|(1 - \gamma(p))\frac{U^1 - U^2 - (\delta_1 - \delta_2)}{U^2 - U^3 - (\delta_2 - \delta_3)}\right|, \tag{12}$$

where

$$\gamma(p) \sim \kappa(2^p\epsilon_1 - (2^p + 1)\epsilon_2 + \epsilon_3) \tag{13}$$

and $\kappa = (2^p - 1)^{-1}$. In practice, one use the approximation

$$p \approx \log_2 \left\|\frac{U^1 - U^2}{U^2 - U^3}\right\|, \tag{14}$$

in $(E_h, \| \|)$. The second-order error term ϵ_2 on u_2 (respt. ϵ_1 on u_1) has therefore $2^p + 1$ (respt. 2^p) more impact on p calculation error than the second-order error term ϵ_3 on u_3. It is interesting to note that the "pointwise" extrapolation

$$v_r^2 = \frac{U^1 U^3 - (U^2)^2}{U^1 - 2U^2 + U^3} \quad \forall x \in M_1$$

in (11) is very sensitive to numerical perturbation, because numerical perturbations are amplified by a factor $(U^1 - 2U^2 + U^3)^{-1}$. Aitken formula is therefore useless numerically unless the perturbation error δ_i is pointwise much less than the asymptotic order of convergence h^p.

A main attribute of RE is its intrinsic simplicity. We are going now to consider its application to CFD with practical examples.

2.2. Application of convergence order approximation and Richardson extrapolation to CFD

Let us consider a general CFD code for steady flows. We denote loosely U one of the flow field output provided by this CFD code. It can be one of the component of the flow speed, the pressure, etc. We suppose that we have three solutions $U^i \in E_i, i = 1..3$ with increasing accuracy on three different grids. To start with, we assume that the corresponding meshes M_i are embedded, in such way that the size of each elements or cells is halfed from M_i to $M_{i+1}, i = 1, 2$. Let I_i: $(E_i, \| \|) \to (E^0, \| \|)$ be an operator that interpolates these flow fields from M_i to a common mesh M^0. Let us assume that this interpolation has a convergence order higher than the consistency error of the CFD code. This hypothesis is critical since, as we have seen before, extrapolation formula may amplify numerical perturbations. In the simplest case, where one can project U^2 and U^3 into $M^0 \equiv M_1$, there are no interpolation errors. Of course, with cell centered finite volume (FV) approximation or with staggered grids in FD, this idealization is not true in practice. We will denote $\tilde{U}^i = I_i[U^i]$ the interpolated field M^0. Further if one seeks an improved solution, then one should look for a mesh M^0 finer than any of all $M_i, i = 1..3$. Then one can apply RE to the family of flow solutions $\tilde{U}^i$.

The asymptotic order of convergence of the solution process may not be known rigorously. The asymptotic formula (14) provides a numerical approximation of p depending on the norm. One can then apply the extrapolation formula (4) on M^0. It is well known that this technique might be applied to practice some code verification [19]. However we are going to show that RE may not work well in CFD, even if the CFD code has been verified properly [13,17,19].

Further, we may adopt a slightly more general point of view and proceed with an arbitrary sequence of grid solution with uniformly decreasing space step $h_1 > h_2 > h_3$. Since the grids are no longer matched uniformly, we use systematically a high order interpolation method on a fine mesh M^0 of space step less than h_3.

In particular, the approximation of the convergence order p is taken to be a solution of the nonlinear equation

$$\frac{h_1^p - h_2^p}{h_2^p - h_3^p} = \frac{\|U^1 - U^2\|}{\|U^2 - U^3\|}. \tag{15}$$

We are going to consider two different codes for the steady state, 2D laminar lid-driven square cavity flow with the Reynolds number, Re, in the range of 20–1000. This is a well-established test case in the literature [12,18]. Our first code denoted $C_{\omega-\psi}$ is based on the FD approximation of the 2D vorticity ω – stream function ψ formulation of the incompressible Navier–Stokes equation with either central FD for the convective term or first-order upwinding [18]. The mesh is squared and regular. We avoid a singularity of h^{-1} order at the end points of the sliding side of the unit square cavity by choosing a speed at the sliding wall as in [2]

$$\frac{\psi}{\partial n} = -x^2(1 - x^2). \tag{16}$$

Formally this code provides second-order accuracy, except that in practice neither the vorticity or the stream function are in $C^4((0,1)^2)$ which is required to justify Taylor expansion truncation errors.

The steady solution in $C_{\omega-\psi}$ is reached by first-order time stepping with explicit Euler Scheme. The stop criterion is

$$\|\rho\|_\infty < h^{3.5}, \tag{17}$$

where h is the space step and ρ the residual for the vorticity equation. Post-processing with RE uses spline interpolation, which satisfies our accuracy requirement.

Our second code, denoted C_{v-p}, is a FV code with centered cells for the 2D Navier–Stokes equation written in velocity–pressure formulation. The convective terms are approximated by first-order upwind or second order central differences depending on the local cell Reynolds number. For a detailed description of the computational procedure, we refer to [25,26]. In principle this code has varying order $p \in (1,2)$ of accuracy depending on the local cell Reynolds number. A second version of this code uses second-order upwinding everywhere for the convective terms. We have confirmed the well-known fact that this FV code handles better the singularity at the corners than $C_{\omega-\psi}$. Interpolation in post-processing is based on the Lagrangian interpolation with either 9-point or 16-point formula properly adjusted near the wall. It was checked carefully that the interpolation procedure used uphold the accuracy of the CFD solution.

Both Navier–Stokes codes has been verified in the following sense:

- Using forcing functions and/or exact divergent free analytical solution, we have checked that the discretization errors is what we can expect from the theory.
- The programming errors that can affect the numerical answer, have been removed by checking with independent coding the discrete residual output.
- We verified that the code output is consistent on several type of hardwares and robust with respect to parametric input such as space step or Reynolds number in the present range of investigation.
- Roundoff errors and robustness of iterative solvers have been double checked by comparing single and double precision, and by studying the impact of stop criterium.

We first analyze the convergence properties of the $C_{\omega-\psi}$ code. Figs. 1–3 show the convergence order of the solution and its RE solution for Reynolds number 20, 100 and 400. We have generated a sequence of grid solutions with a regular grid of space step $h_j = 0.1/j$ with $j = 1..10$. Let us denote now U^j to be the corresponding grid solutions, and let $\tilde{U}^j$ be the interpolated grid solution on the finest grid via spline interpolation. We let also $\hat{U}^j$ be the projection of U^{2j} into the coarser grid of space step h_j.

In these figures, we compare the order estimate from the solution of the nonlinear Eq. (15) with interpolated grid solution on the finest grid $\tilde{U}^j$, i.e., p_{j+1} solution of

$$\frac{h_j^p - h_{j+1}^p}{h_{j+1}^p - h_{j+2}^p} = \frac{\|\tilde{U}^j - \tilde{U}^{j+1}\|}{\|\tilde{U}^{j+1} - \tilde{U}^{j+2}\|} \tag{18}$$

and the order estimated based on the solution projected on the coarse grids, i.e.,

$$p_{j+1} = \log\left(\frac{\|U^j - \hat{U}^j\|}{\|U^{j+1} - \hat{U}^{j+1}\|}\right) / \log\left(\frac{h_j}{h_{j+1}}\right). \tag{19}$$

The first two results with $Re = 20$, and $Re = 100$ (see Figs. 1 and 2), are obtained with second-order central differences for the convective terms. The solution should be formally second order. For the stream function, we observe that the convergence order increases slowly to its asymptotic limit 2. We observe that

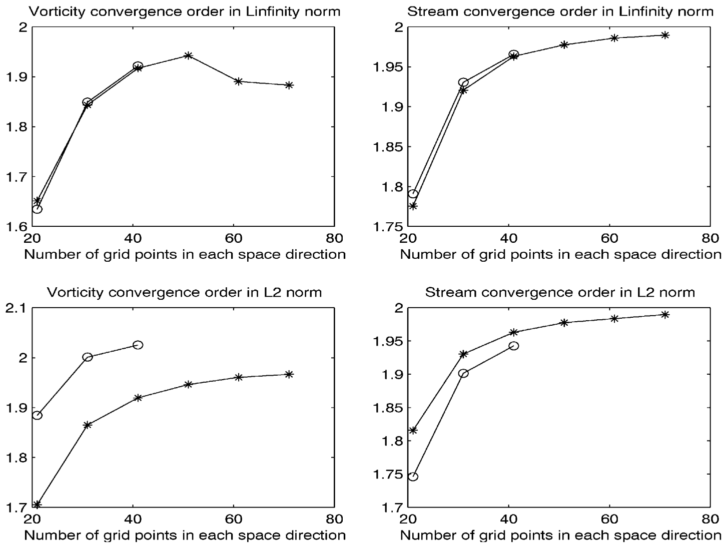

Fig. 1. Convergence order approximation for $C_{w-\psi}$ code with $Re = 20$: ○, curve for coarse grid projection solution; *, curve for fine grid interpolation solution.

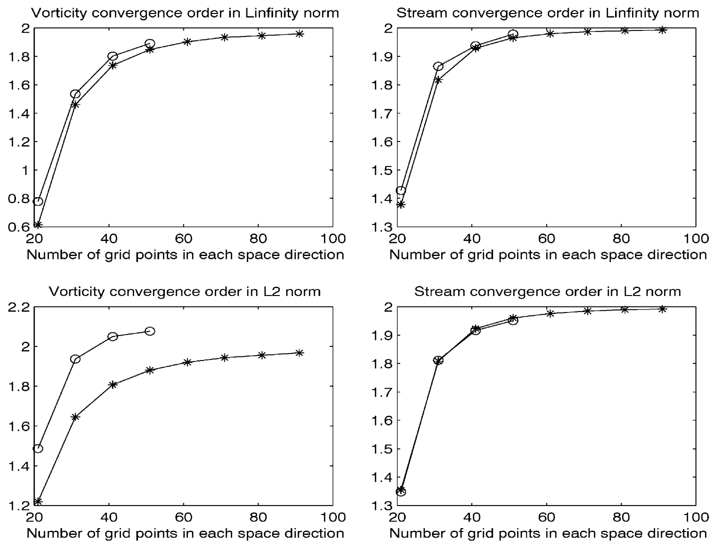

Fig. 2. Convergence order approximation for $C_{w-\psi}$ code with $Re = 100$: ○, curve for coarse grid projection solution; *, curve for fine grid interpolation solution.

the spline interpolation errors decreases with the space step and has little effect on the convergence order for the smallest space steps. For the vorticity, the situation is more complicated because of the singularity at the corners of the sliding wall. As a matter of fact the difference between L_∞ and L_2 errors revealed this local singularity (see Figs. 1 and 2). We have also an overestimation of the convergence order of the grid solution projected on the coarse grid.

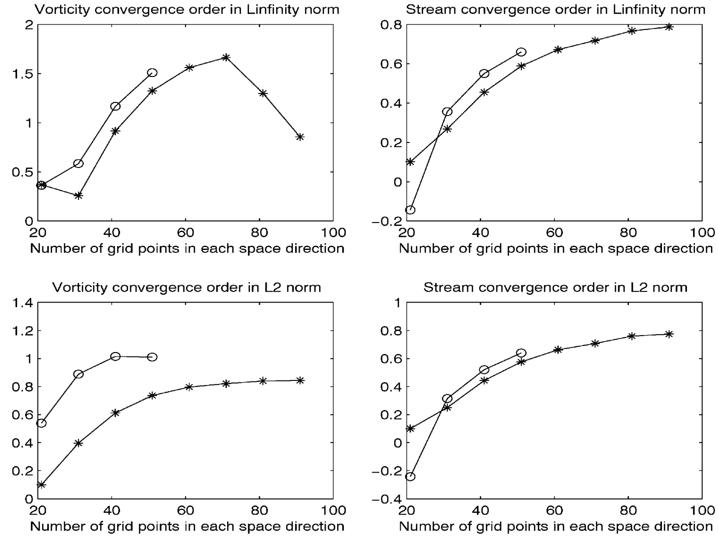

Fig. 3. Convergence order approximation for $C_{w-\psi}$ code with $Re = 400$: ○, curve for coarse grid projection solution; *, curve for fine grid interpolation solution.

The third result with $Re = 400$ shown in Fig. 3, was obtained with first-order upwinding for the convective terms. The convergence is extremely poor for the coarse grid solutions, $j = 1..3$. For the stream function. We observe that the convergence order increases slowly to its asymptotic limit of 1. For the vorticity, the situation is rather interesting because the singularity at the corners of the sliding wall is worse than in two previous cases. Heuristically the code behaves like first-order outside the boundary layers where the diffusion is negligible, and second order where the error on the diffusion term is dominant. This explain why in L_∞ norm the convergence order is a nonmonotonic curve. As a matter of fact for low grid resolution the error in the boundary layer is dominant and the error on viscous term can be worse than the error on convective terms.

A more detailed analysis can be done by plotting the convergence order of our solution using the asymptotic formula (10) in a pointwise manner

$$p(x) = \log_2 \frac{|\tilde{U}^j - \tilde{U}^{2j}|}{|\tilde{U}^{2j} - \tilde{U}^{4j}|} \quad \forall x \in M^0. \tag{20}$$

We observe that the formula is singular at grid points where $\tilde{U}^{2j} = \tilde{U}^{4j}$ within roundoff error. This is not in contradiction with a convergence in L_2 or possibly L_∞ norm. We therefore should not take into account the evaluation of $p(x)$ at grid points where $\tilde{U}^{2j}(x) - \tilde{U}^{4j}(x) \approx h_j^3$. The surface plots of $p(x)$ in Fig. 4 shows two curves starting from the wall where this cancellation phenomenon destroys the validity of p order approximation. The slide $y = 0.2$ in Fig. 5 of the surface plot of convergence order of Fig. 4 exhibits the typical singularities $\sim 1/x$ of the convergence order corresponding to these two curves.

The slide $y = h$ in Fig. 6 is the closest to the sliding wall. It shows two additional zones near the end points where the convergence order is obviously less than satisfactory. It therefore confirms the poor performance of the code $C_{\omega-\psi}$ in the neighborhood of the corner of the sliding side for large Reynolds number. The singularity of the vorticity solution at the corner is therefore not totally removed by the wall condition (16). RE can be applied to our interpolated grid solution, as shown in Figs. 10–12. To analyze the result is not straightforward, because we have moderated size of data with inadequate accuracy and because the convergence properties of the $\omega - \psi$ code are peculiar along the sliding wall. We will comment on these results in combination to least square method results in Section 4.2. We now consider the FV code

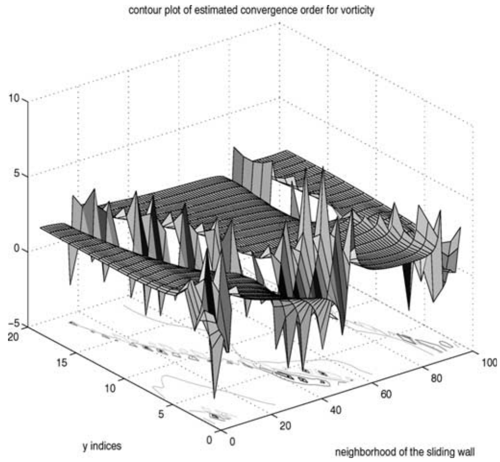

Fig. 4. Surface plot of space dependent convergence order approximation for $C_{w-\psi}$ code with $Re = 100$.

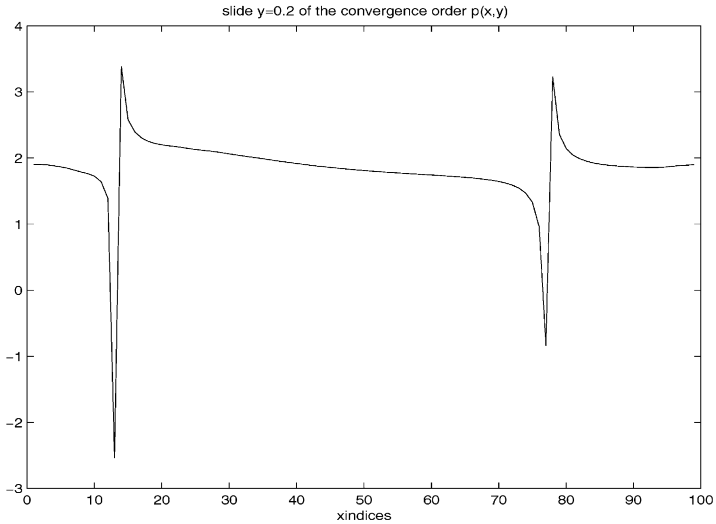

Fig. 5. Slide $y = 0.2$ of space dependant convergence order approximation for $C_{w-\psi}$ code with $Re = 100$.

employing the velocity-pressure formulation, which gives more robust solutions and leads to easier interpretation of Richardson extrapolation accuracy.

We refer to [27] for an extensive study that provides the sensitivity of the result with respect to the choice of the norm and the effect of the accuracy of the interpolation formula. One can conclude from these experiments that the FV approach is more robust to singularities than the FD method. Nevertheless, one still observes a lower order of convergence near the corner of the sliding wall for $Re = 1000$. According to [27], for most of the cases with low Reynolds number, it can be shown that second order RE reduces the error in discrete L_1, L_2 and L_∞ norms. For higher Reynolds number such as $Re = 1000$, RE fails to improve the error in the L_2 norm when the coarse grid is not fine enough. Overall, RE can improve the order of accuracy but not consistently.

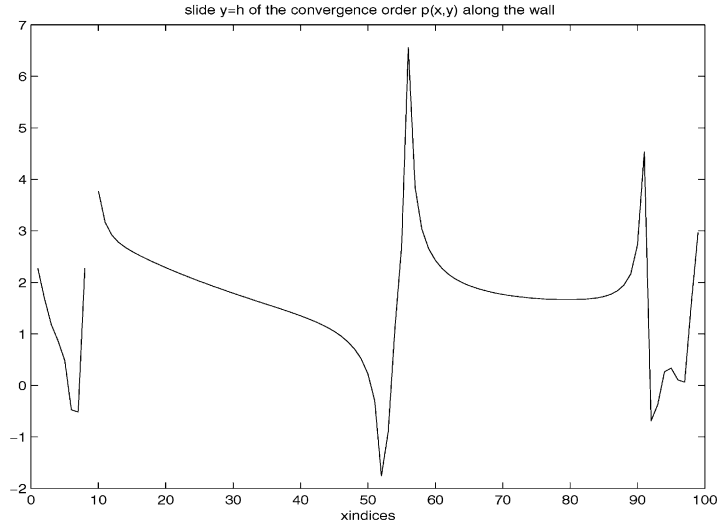

Fig. 6. Slide $y = 0.01$ of space dependant convergence order approximation for $C_{w-\psi}$ code with $Re = 100$.

From these experiences and the basic theory of RE in Section 2.1, we conclude that there are many factors affecting the performance of RE. First of all the convergence order of a solution process is usually space dependent. It is clear for example in the case of the hybrid treatment of convective terms depending on the local cell Reynolds number. It is also well known that grid refinement may lead to grid solution that cannot be compared uniformly as in the RE definition used so far. Let us mention also that for pseudo-spectral approximation used in CFD, the convergence order is not even algebraic: in other word the convergence order increases with the grid refinement [23], There are therefore no way to justify RE with constant weights formula in this context. Further, in practice with FE or FV, it is not clear that the convergence order is satisfied because one often cannot afford a mesh fine enough to get close to asymptotic estimates. This is typically the case in 3D computations. It is therefore desirable to extend the concept of RE to solve CFD problem. We present in the following section a new extrapolation method that has been designed to solve some of the limitations of RE.

3. Least square extrapolation for numerical functions

We go back to the mathematical framework of Section 2.1, and set $E = L_2(0,1)$. Let u be a function of E. Let v_h^1 and v_h^2 be two approximations of u in E such that $v_h^1, v_h^2 \to u$ in E as $h \to 0$. A consistent linear extrapolation formula writes formally $\alpha v_h^1 + (1-\alpha)v_h^2$. As a matter of fact, this formula must be α independent when $v_h^1 = v_h^2 = u$. In p order RE the α function is a constant. We adopt here a more general point of view than RE: we look for a convergence order that is space dependent. Further, we do not require that v_h^2 is a uniformly more accurate approximation of u than v_h^1.

We define then the following problem:

P_α: Find $\alpha \in \Lambda(0,1) \subset L_\infty$ such that $\alpha v_h^1 + (1-\alpha)v_h^2 - u$ is minimum in $L_2(0,1)$.

The space of function $\Lambda(0,1)$ will be defined later on. Let us note, first that if $1/(v_h^1 - v_h^2)$ is in $L_\infty(0,1)$, we get

$$\alpha = \frac{u - v_h^2}{v_h^1 - v_h^2}. \tag{21}$$

This problem has therefore a unique solution in $L_2(0,1)$. Second, if $v_h^1 - v_h^2$ vanishes, we can approximate then v_h^i by a w_h^i function in $L_2(0,1)$ such that

$$w_h^i - u_h^i = O(h^q) \tag{22}$$

and

$$1/(w_h^1 - w_h^2) \in L_\infty(0,1). \tag{23}$$

In order to solve the modified problem P_α with asymptotically equivalent data (w_h^1, w_h^2). We take then q a positive integer such that $q > p$ where p is the expected order of convergence of v_h as $h \to 0$. We get then

$$\alpha = \frac{u - w_h^2}{w_h^1 - w_h^2} \tag{24}$$

and $\alpha \in L_2(0,1)$.

We observe further that we need to find only a rough approximation of α function because the convergence order of v_h approximation is multiplied by the convergence order of α approximation as stated in the following lemma.

Lemma 1. *If $\alpha_M - \alpha = 0(M^{-1})$ as $M \to \infty$ and $v_h^1 - v_h^2 = 0(h^p)$ then*

$$u = \alpha v_h^1 + (1-\alpha)v_h^2 + O(h^p) \times O(M^{-1}). \tag{25}$$

Proof. A simple computation gives

$$u - (\alpha_M v_h^1 + (1-\alpha_M)v_h^2) = (\alpha - \alpha_M)(v_h^1 - v_h^2). \qquad \square \tag{26}$$

Finally the arithmetic cost to find an approximation of a solution of P_α must be in practice much lower than the cost of a "fine grid solution", v_h^3 that will provide the same order of accuracy than the extrapolated function. $\Lambda(0,1)$ must therefore be a finite linear space that can be represented by few basis functions.

In the present work, we set $\Lambda(0,1)$ to be the space of α functions

$$\alpha = \alpha^0 + \alpha^1 \cos(x\pi) + \sum_{j=1..M} \alpha^j \sin((j-1)x\pi). \tag{27}$$

with $\alpha^j, j = 0..M$ reals.

We will first review some basic properties of $\Lambda(0,1)$ in the approximation of $L_2(0,1)$ functions.

Lemma 2. *Let α be in $L_2(0,1)$ Let $x_j = j/M$* be a *regular discretization of $(0,1)$. There is a unique trigonometric polynomial*

$$\alpha_M = \alpha^0 + \alpha^1 \cos(x\pi) + \sum_{j=1..M} \alpha^j \sin((j-1)x\pi)$$

that interpolates α on x_j. α_M converges to α in $L_2(0,1)$ as $M \to \infty$. Further, if $\alpha \in C^2(0,1)$, the convergence $\alpha_M \to v$ is pointwise and of order M^{-2} in (0,1) and M^{-3} away from the end points.

Proof. We observe that $\alpha^0 + \alpha^1\cos(x\pi)$ interpolates α at $x = 0$ and $x = 1$. Let us define f a periodic function in L_2 such that $f(x) = \alpha(x\pi) - \alpha^0 - \alpha^1\cos(x\pi)$ on (0,1) and $f(x) = -f(x)$. for $x \in (-1,0)$. The result comes from the classical Fourier approximation theory applied to f (see [10] and its references). $\square$

We can now derive an approximation for least square approximation of $L_2(0,1)$ functions.

Lemma 3. *Let α be in $L_2(0,1)$. Let $x_j = j/N$ be a regular discretization of $(0,1)$. Let M be an integer such that $M \ll N$. There is a unique trigonometric polynomial $\alpha_M = \alpha^0 + \alpha^1 \cos(x\pi) + \sum_{j=1..M} \alpha^j \sin((j-1)x\pi)$ that minimizes the discrete L_2 norm*

$$\sum_{j=0..N} (\alpha(x_j) - \alpha_M(x_j))^2, \tag{28}$$

where α_M converges to α in $L_2(0,1)$ as $M \to \infty$ while the ratio M/N stays constant and is less than one. Further, if $\alpha \in C^2(0,1)$, the convergence $\alpha_M \to v$ is pointwise and of order M^{-2} in (0,1) and M^{-3} away from the end points.

Proof. First we construct from the set of basis function $b_0(x) = 1$, $b_1(x) = \cos(x\pi)$, $b_j(x) = \sin((j-1)x\pi)$, $j = 2..M$ an orthogonal basis using the Gram–Schmidt process as follows. Let us denote this new basis $\{e_i, i = 0..M\}$. Since $\{b_j, j = 2..M\}$ is already an orthogonal family, we set $e_j = b_j$, $j = 2..M$. Then we set

$$e_0 = b_0 - \sum_{j=2..M} (b_0, e_j) e_j / \|e_j\| \tag{29}$$

and

$$e_1 = b_1 - (b_1, e_0) e_0 / \|e_0\| - \sum_{j=2..M} (b_1, e_j) e_j / \|e_j\|. \tag{30}$$

The least square approximation of α is then its L_2 projection into $\Lambda(0,1)$ using the orthogonal basis e_j. The convergences properties follow from Lemma 2. □

We have now a solution to the approximation problem P_α or its modified analog if we have possibly to modify locally the v_h^i function at neighborhood of points where $v_h^1 - v_h^2$ cancels as in Eqs. (22) and (23).

From Lemmas 1 and 3, we have

Theorem 1. *If $u, v_h^i, i = 1,2 \in C^1(0,1)$, $\frac{1}{v_h^1 - v_h^2} \in L_\infty(0,1)$ and $v_h^2 - v_h^1 = 0(h^p)$ then $\alpha v_h^1 + (1-\alpha) v_h^2$ is an $0(M^{-2}) \times 0(h^p)$ approximation of u.*

We observe that special care must be done if

$$v_h^1 - v_h^2 \ll u - v_h^2, \tag{31}$$

in some set of nonzero measure Ω_S. This situation is typical of insufficient local convergence of CFD codes – see Fig. 4 for example. Further, one expect that the least square approximation $w_h = \alpha v_h^1 + (1-\alpha) v_h^2$ of u with $\alpha \in C^0(0,1)$ will be a poor approximation in Ω_S. However, this is consistent with the fact that (31) means that the convergence of v_h at points in Ω_S fails! One can constraint the least square approximation problem by imposing that the α approximation stays bounded by q. These outliers points will therefore not affect globally the least square extrapolation as long as we impose α to be a bounded function independently of h. At grid points where (31) holds, the extrapolated function will coincide asymptotically with v_h^2 which is the best that one can get under such condition.

In order to define a more robust approximation of u, we introduce the second problem as follows:

$P_{\alpha,\beta}$: Find $\alpha, \beta \in \Lambda(0,1)$ such that

$$\alpha v_h^1 + \beta v_h^2 + (1 - \alpha - \beta) v_h^3 - u \tag{32}$$

is minimum in $L_2(0,1)$.

If one can partition $(0,1)$ into two overlapping subset $\Omega_1 \cup \Omega_2 = (0,1)$ of nonzero measure intersection, such that $1/(v_h^1 - v_h^3)$ is in $L_\infty(\Omega_1)$ and $1/(v_h^2 - v_h^3)$ is in $L_\infty(\Omega_2)$, then it can be shown that this problem has a solution in $L_2(0,1)$. Although uniqueness is no longer guaranteed, in practice, since we only look for a low order least square approximation, we do reach a unique solution. While we can use a singular value decomposition method (SVD), as recommended in [21] to account for the fact that the linear system can be either over- or under-determined. It may not be desirable because SVD requires much more arithmetic operations than directly solving the normal set of equations when $M \ll N$. In practice, it is unlikely that $v_h^1 - v_h^3 \ll u - v_h^3$ and $v_h^2 - v_h^3 \ll u - v_h^3$ in some set of nonzero measure. In the neighborhood of grid points where this is not the case, there is no convergence of v_h^i, and one can locally modify these functions in order to retrieve v_h^3 as the best fit.

Another type of improvement to the least square approximation presented so far may come from the domain decomposition point of view. It is common to introduce adaptivity to approximate functions that have strong gradient in space. Since we use trigonometric polynomial on regular grids to approximate our weight functions, we rather introduce an (adaptive) overlapping domain decomposition approach of our least square extrapolation problem P_α or $P_{\alpha,\beta}$ in order to possibly increase the accuracy locally. We proceed as follows. Let (a_j, b_j), $j = 1..n$ be a set of overlapping intervals such that

$$a_1 = 0 < a_2 < b_1 < a_3 < b_2 < \cdots < b_n = 1. \tag{33}$$

We assume that the end point of (a_j, b_j) coincide with some grid points x_k, $k = 0..N$. Let M_j, $j = 1..n$ be such that $\sum_{j=1..n} \mathrm{Card}(M_j) \ll N$. The $\Lambda(0,1)$ space is then the set of functions

$$\alpha_M \equiv \{\alpha_j,\ \text{for } x \in (a_j, b_j)\}_{j=1..n},\ y = (x - a_j)/(b_j - a_j), \tag{34}$$

such that

$$\alpha_j = \alpha_j^0 + \alpha_j^1 \cos(y\pi) + \sum_{k=1..M_j} \alpha_j^k \sin((k-1)y\pi). \tag{35}$$

This overlapping domain decomposition technique is standard and can have variant with different type of matching conditions and overlap.

Now let us focus on a difficulty that is most relevant to the application side. In practice, we work with *grid functions* solution of discretized PDE problems as in Section 2.2. So we will denote as before $U^i \in E_i$, $i = 1..3$ three grid functions defined on M_i grids, that are approximations of a grid function $U \in E^0$. We suppose that the corresponding grid M^0 has constant space step h^0. We do not suppose that the M_i grids have regular space step, neither we need to suppose that the sequence of U^i functions has increasing order of accuracy. We denote then as in Section 2.2 $\tilde{U}_i = I_i[U_i]$, the grid function on M^0 obtained by high order interpolation. The least square extrapolation problems are then, respectively

P_α: Find $\alpha \in \Lambda(0,1) \subset L_\infty$ such that

$$\alpha\tilde{U}^1 + (1-\alpha)\tilde{U}^2 - U \tag{36}$$

is minimum in $L_2(M^0)$.

and

$P_{\alpha,\beta}$: Find $\alpha, \beta \in \Lambda(0,1)$ such that

$$\alpha\tilde{U}^1 + \beta\tilde{U}^2 + (1-\alpha-\beta))\tilde{U}^3 - U \tag{37}$$

is minimum in $L_2(M^0)$.

The least square procedure to solve these problems is therefore standard and as mentioned before, one can use SVD to diagnose and overcome possible numerical difficulties. Furthermore, several statistical tools are available to control the accuracy [22]. However the choice of the interpolation tool is critical. It is desirable to use a high order Lagrange interpolation method or spline. To anticipate the results of the following section, we will see that the interpolation procedure may obey some constraint as smoothness and that spline interpolation is preferable in the context of PDEs.

Let us illustrate the numerical accuracy and sensitivity to perturbation of our least square extrapolation method. We consider two examples with the C^0 function

$$v(x) = x \text{ in } (0, \pi/2), \quad \text{and} \quad v(x) = \pi - x, \in (\pi/2, \pi) \tag{38}$$

and the C^∞ function

$$v(x) = \exp(\sin(x)) - 1 + \frac{\pi}{4}x(1-x). \tag{39}$$

For simplicity, we have taken v vanishing at the end points $x = 0, \pi$.

The grid function V_h on the embedded grid $G1 \subset G2 \subset G3$ are given using one of the following asymptotic expansions structures,

$$v_h = v + hv_1 - h^2v_2 + h^4v_3 \tag{40}$$

and

$$v_h = v + h^2v_1 - h^4v_2, \tag{41}$$

with v_i of order one in $(0, \pi)$. The function v_i is chosen such that the cancellation phenomenon such as (31) occurs at some points. Further, we apply extrapolation to noisy data, $\tilde{v}_h = v_h + r_h$, with r_h being a small random perturbation function. In Fig. 7, we show limits and success of the least square extrapolation presented here. In this numerical experiment, the three grids $G_i, i = 1..3$ are embedded with increasing number of grid points 13, 25, and 49. The final result is the mean of the error in L_2 norm, given on G_4 that has 97 grid points with order ten computations with different arbitrary random noise of given fixed size from 10^{-1} down to 10^{-5}. Spline is used to interpolate the grid function data on the fine grid G_4. Each weight function α and/or β is computed using $M = 5$ modes only. We observe that the two-level and three-level extrapolation fail to improve the second order RE, for nonsmooth function as (38) when the asymptotic expansion v_h converges to v at second order as shown in Fig. 7(a). This is mainly due to the fact that the weight functions are computed in the Fourier space, and the Gibbs phenomenon affects the second-order accuracy. However, we see some improvement for data corresponding to first-order convergence, i.e., (40) expansion – see Fig. 7(c). For smooth functions as in (39), as shown in Figs. 7(b) and (d), the three-level extrapolation approximation always gives good result, as opposed to the two-level method that can suffer from the cancellation phenomenon. In Figs. 7(e) and (f), we consider the case of varying order of convergence, i.e., we cut off v_1 in (40) to be 0 in $(0, \pi/2)$. We observe that the least square extrapolation is definitively an improvement on fixed order RE in this situation. In all cases our least square extrapolation method give better result than first- or second order RE, whatever is the size of the noise r_h added randomly to each level grid function V_h, whatever is the convergence order of v_h to v.

So far, we have restricted our presentation to one-dimensional (1D) approximation. The extension to multidimensional problem is straightforward on tensorial product of grids. However this in principle limits us to problems with very simple geometry. In fact our least square extrapolation method should be generalized naturally to complex geometry problem by fictitious domain technique [8]. As a matter of fact, it is straightforward to embed any general domain $\Omega \subset \mathbb{R}^n$ into a box $(0,1)^n$ after an eventual rescaling of space variables. We impose then M^0 to be a tensorial product of 1D grid with constant space step in each space direction of (0,1).

We extend then the functions $\tilde{U}^i(x)$ and U for $x \in \Omega \cap M^0$ to grid functions defined on M^0 using technique similar to [7] or [14]. Once again using domain decomposition, one can do refinement in subdomains

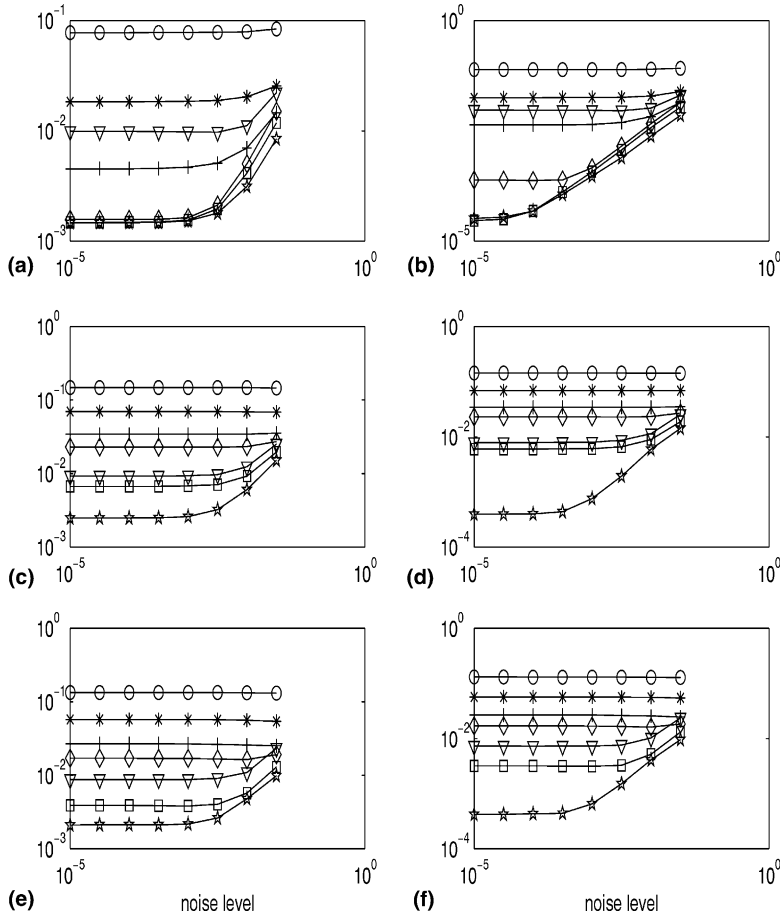

Fig. 7. Sensitivity and accuracy experiment of extrapolation methods for given analytical functions. ○, for G1; *, for G2; +, for G3; ▽, for R1; ◇, for R2; □, for LS1; ☆, for LS2.

with regular grids. Another promising generalization is to take advantage of body fitted meshes generated by PDEs [28,29], since the Fourier expansion technique is insensitive to change of variables as long as it is a smooth transformation.

We are going to extend now our least square extrapolation method to construct approximate solutions of PDE problems in square domains.

4. Least square extrapolation for PDEs

In classical RE methods applied to PDEs solution, one supposes that the order of convergence of the sequence of functions u_i, $i = 1..3$ is known. The fact that these functions are solution of a discrete approximations of the PDE problem is never used directly. On the contrary, our criterion to determine the weights in the present least square extrapolation formula is to approximate the PDE problem itself. So we intend to approximate a fine grid solution of the approximation problem rather than to approximate directly the exact solution as in RE.

4.1. Computational algorithm

Let us denote formally

$$L[u] = f \quad \text{with } u \in (E_a, \| \, \|_a) \text{ and } f \in (E_b, \| \, \|_b), \tag{42}$$

a linear PDE problem and its numerical approximation,

$$L_h[U] = f_h \quad \text{with } U \in (E^h_a, \| \, \|_a) \text{ and } f_h \in (E^h_b, \| \, \|_b), \tag{43}$$

parameterized by a mesh step h.

Suppose that we have the stability estimate

$$\|U\|_a \leqslant Ch^s(\|f_h\|_b), \tag{44}$$

with s real but not necessarily positive. If the truncation error, $\rho_h = L_h[U - u_h]$ satisfies the estimate $\|f_h\|_b = 0(h^q)$ with $q > 0$, then the numerical method is $s + q$ order. Since the operator is linear, at first sight, both q order and $s + q$ order RE formula should give asymptotically the same improved solution. In practice, the estimate (44) may not be optimal. One therefore should rather use the q order Richardson extrapolation formula that minimizes the residual.

Let us restrict ourselves to two-point boundary value problems in (0,1). We recall that we intend to get an improved approximation of the PDE's solution on the fine grid M^0 of space step noted loosely as h. We extend our least extrapolation problem formulation for PDEs problems as follows.

P_α: Find $\alpha \in \Lambda(0,1) \subset L_\infty$ such that

$$\alpha L_h[\tilde{U}^1] + (1-\alpha)L_h[\tilde{U}^2] - f_h \tag{45}$$

is minimum in $L_2(M^0)$.

and

P_α: Find $\alpha, \beta \in \Lambda(0,1)$ such that

$$\alpha L_h[\tilde{U}^1] + \beta L_h[\tilde{U}^2] + (1-\alpha-\beta)L_h[\tilde{U}^3] - f_h \tag{46}$$

is minimum in $L_2(M^0)$.

We are going to focus our study in this paper on the practical use of this new methodology. The mathematical theory for given classes of linear operators need to be developed in future work. We remark that it is essential that the interpolation operator gives a smooth interpolant. As a matter of fact, if L_h has second-order derivatives, we will use spline interpolant that preserve $C^2(0,1)$ continuity. For conservation laws, one may require that the interpolation operator satisfies the same conservation properties. For chemical problems, one may require that the interpolant preserve the positivity of species.

For elliptic problems, it is convenient to post-process the interpolated functions $\tilde{U}^i$, $i = 1..3$ by few steps of the relaxation scheme

$$\frac{V^{k+1} - V^k}{\delta t} = L_h[V^k] - f_h, V^0 = \tilde{U}^i, \tag{47}$$

with appropriate artificial time step δt. As a matter of fact this procedure dampens the high frequencies that are introduced by the interpolation and significantly amplified by derivatives operator. This filtering process happens to be critical in the application of the least square method to the Navier–Stokes equations, as will be discussed later.

The solution process of P_α and/or $P_{(\alpha,\beta)}$ can be decomposed into three consecutive steps. First, interpolation from G_i, $i = 1..3$ to M^0 that has a number of arithmetic operations proportional to $\text{Card}(M^0)$, i.e., the number of grid points of M^0. Second, the evaluation of the residual on the fine grid M^0, that has the same asymptotic order of arithmetic operations. Third the solution of the linear least square problem with M unknowns. If we keep M of the same order as $\text{Card}(M^0)^{1/3}$, and use a standard direct solver for symmetric system to solve the normal set of equations, the arithmetic complexity of the overall procedure is still of order $\text{Card}(M^0)$, i.e., it is linear.

To push further the practical use of our method, we observe that it can be generalized in a straightforward way to nonlinear PDE problem, via a Newton-like loop. To be more specific, let us denote

$$N[u] = f, \tag{48}$$

the nonlinear problem. A Newton scheme results in a sequence of linearized problem denoted here,

$$J(u)[v] = g. \tag{49}$$

Let us consider first the two-level extrapolation case. We will solve a sequence of P_α problems

P_{α^k}: Find $\alpha^k \in \Lambda(0,1) \subset L_\infty$ such that

$$\alpha^{k+1} J_h(\alpha^k \tilde{U}^1 + (1-\alpha^k)\tilde{U}^2)[\tilde{U}^1] + (1-\alpha^{k+1}) J_h(\alpha^k \tilde{U}^1 + (1-\alpha^k)\tilde{U}^2)[\tilde{U}^2] - g_h \tag{50}$$

is minimum in $L_2(M^0)$, starting from initial condition $\alpha^0 \equiv 0$, until $\|\alpha^{k+1} - \alpha^k\|$ is less than some tolerance number. The convergence of this scheme is not guaranteed, and as usual in Newton like method the initial guess should not be too far from the final solution, i.e., the grid solution on M^0 should not be too far from the grid solution on G2.

The algorithm for the three-level extrapolation case is based on the identical concept. In order to proceed with the evaluation of our method, we now consider two linear and nonlinear examples for which traditional RE does not work properly.

Our least square extrapolation method is coded as an independent program from the original PDE code that produce the grid solution. The algorithm is as follows:

Step 1: compute the spline interpolation of each grid solution onto a common fine grid solution M^0.

Step 2: compute once and for all the set of functions:

$$L[e_i(\tilde{U}^1 - \tilde{U}^2)], \tag{51}$$

for the two-level case, and additionally

$$L[e_i(\tilde{U}^2 - \tilde{U}^3)], \tag{52}$$

for the three-level case, where the e_i are the basis function of Λ.

Step 3: solve the least square problem P_α or $P_{\alpha,\beta}$.

We choose a Fourier expansion for each weight function α and β that has M terms with $M \approx \text{Card}(M^0)^{1/3}$, to keep a linear cost for the complete procedure when the direct solution of the normal set of equations is giving good result. An SVD, if needed will lead however to more intense computation [9]. If the problem is nonlinear, then we repeat steps 1–3 as many Newton iterations are computed. We have used a Matlab implementation of this procedure independent of the code that generates the grid solutions. The numerical experiments thereafter are restricted to 2D problems in square domains. We refer to [6] for previous experiments with a 1D viscous steady Burgers problem that has a boundary layer.

4.2. Numerical evaluation

In all numerical experiments presented below, we will use the following abbreviations: G1 for direct numerical solution, without any extrapolation, on grid $N_1 \times N_1$, G2 for direct numerical solution on grid $N_2 \times N_2, \ldots$, etc., R1 for RE assuming first-order convergence using G2 and G3 data, R2 for RE assuming second-order convergence using G2 and G3 data, LS1 for two-level least square extrapolation using G2 and G3 data, LS2 for three-levels least square extrapolation using G1, G2, G3 data.

In Figs. 8–12, we report on the error of the coarse grid solutions G1, G2 and G3 as well as RE and least square extrapolation versus the exact grid solution on G4 for increasing resolution with G4. the curves corresponding to solutions on coarse grids G1, G2, G3 are processed after being interpolated on the fine grid G4 with spline interpolation. The error variations in these results with respect to the grid size of G4 are caused by projection inaccuracies from spline interpolation. RE as well as least square extrapolation used the data from G1, G2 and G3 that have been obtained *after* spline interpolation on G4. RE and least square extrapolation are therefore providing grid solution directly on G4 that are compared to the exact grid solution on G4.

The matlab conventions are used for label, that is, 's' for squares, 'd' for diamond, 'v' for triangle (down), 'p' for pentagram, 'h' for hexagram.

4.2.1. 2D turning point problem

In this section, we consider the following 2D turning point problem,

$$\epsilon \Delta u + a(x,y)\frac{\partial u}{\partial x} = 0, \quad x \in (0,\pi)^2, \tag{53}$$

$$a(x,y) = x - \left(\frac{\pi}{2} + 0.3\left(y - \frac{\pi}{2}\right)\right). \tag{54}$$

We define this problem in such a way that the transition layer of ϵ order thickness centered on the curve $a(x,y) = 0$, is not parallel to the x or y axis. Therefore, the problem has really a 2D structure. The three levels solutions are provided by a finite difference code that has second order central difference approximation of the diffusion term and first-order upwinding for the convection term. One uses either a direct sparse LU linear or GMRES solver [16,24] since the matrix is nonsymmetric. Because of the discrete maximum principle satisfied by the discrete operator, we avoid creating spurious oscillations. Further, if the grid is not fine enough, one observes that the solution may not be accurate in the transition layer but that

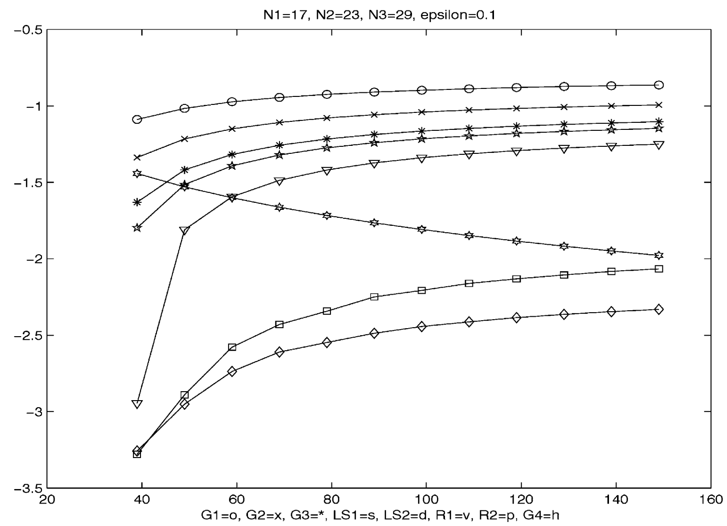

Fig. 8. Application to a turning point problem with $\epsilon = 0.1$. x axis is for the number of grid points N in each space direction for G4. y axis gives in $\log_{10}$ scale the errors in maximum norm. 's' for squares, 'd' for diamond, 'v' for triangle (down), 'p' for pentagram, 'h' for hexagram.

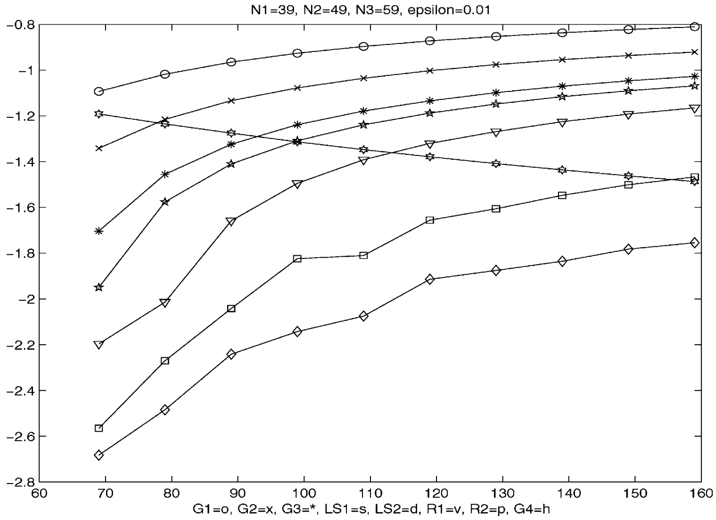

Fig. 9. Application to a turning point problem with $\epsilon = 0.01$. x axis is for the number of grid points N in each space direction for G4. y axis gives in $\log_{10}$ scale the errors in maximum norm. 's' for squares, 'd' for diamond, 'v' for triangle (down), 'p' for pentagram, 'h' for hexagram.

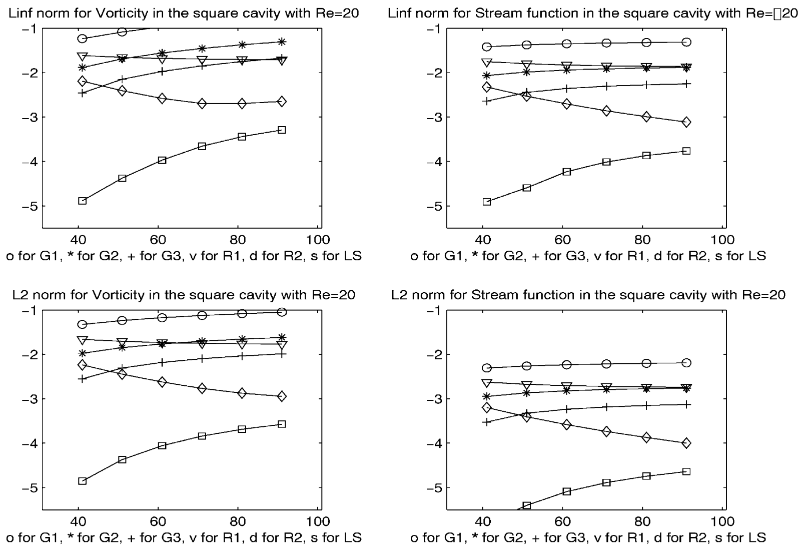

Fig. 10. Application to the lid-driven cavity problem and $C_{w-\psi}$ code with $Re = 20$. x axis is for the number of grid points N in each space direction for G4. y axis gives in $\log_{10}$ scale the relative errors in maximum norm and L_2 norm.

the solution can still have first-order accuracy outside the layer. It can be observed that the solution accuracy behaves as second order in L_∞ norm if the grid is not fine enough because the error on the diffusion term dominates. Figs. 8 and 9 report on the accuracy of the two-level and three-level least square extrapolation versus RE assuming either first- or second-order convergence. The errors are given in L_∞ norm. The curve with hexagram signs gives an accurate estimation of the discrete solution error between the exact

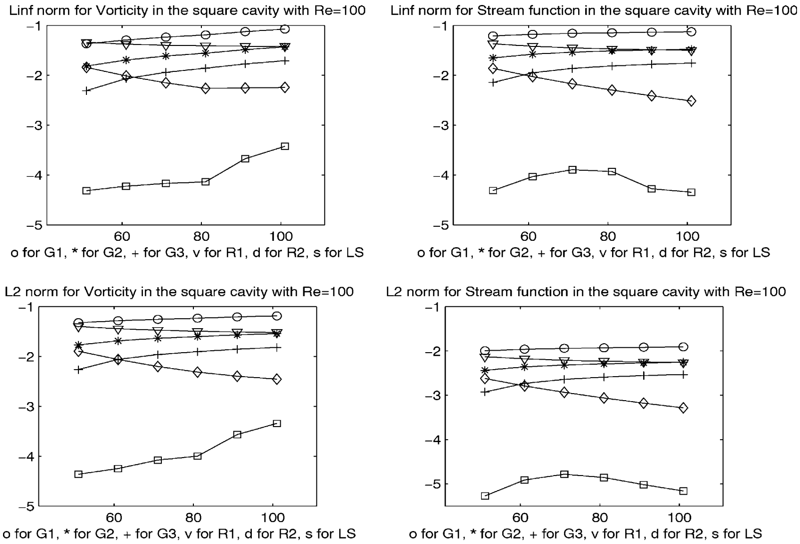

Fig. 11. Application to the lid-driven cavity problem and $C_{w-\psi}$ code with $Re = 100$. x axis is for the number of grid points N in each space direction for G4. y axis gives in $\log_{10}$ scale the relative errors in maximum norm and L_2 norm.

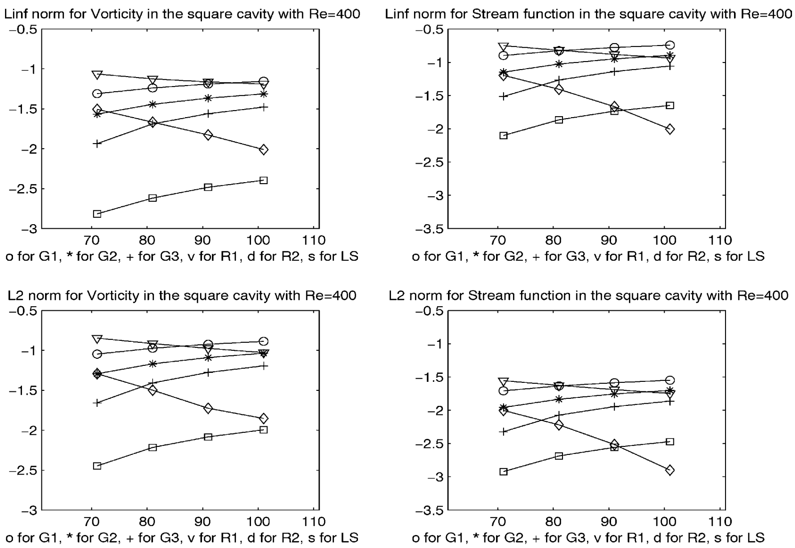

Fig. 12. Application to the lid-driven cavity problem and $C_{w-\psi}$ code with $Re = 400$. x axis is for the number of grid points N in each space direction for G4. y axis gives in $\log_{10}$ scale the relative errors in maximum norm and L_2 norm.

grid solution on G4 versus the exact continuous solution of the turning point problem. This estimate uses as an approximation of the exact solution the projection on G4 of the embedded G5 grid solution of space step half of G4 space step.

The number of Fourier modes in the approximation of the weight α, β is 4 in each space direction. We observe that for both cases $\epsilon = 0.1$ and $\epsilon = 0.01$ in Figs. 8 and 9, R1 gives better results than R2. This is an indication of the fact that the transition layer is not under-resolved.

We observe in Fig. 8 with $\epsilon = 0.1$, and modest base grid sizes, namely, $N_1 = 17$, $N_2 = 23$, $N_3 = 29$, meaning that we have on average only one or two grid points in the transition layer for the G3 solution, our least square is as accurate as the G4 grid solution. As a matter of fact the error for the G4 solution labeled with hexagram stays above the error for the least square approximations of G4 solution (see the curves with square and losange labels). This is still true when the RE fails for $N \geqslant 70$. The least square extrapolation also gives satisfactory results in Fig. 9, where $\epsilon = 0.01$, $N_1 = 39$, $N_2 = 49$, $N_3 = 59$, but R1 predicts the grid solution on G4 with an error less than or equal to the error with the exact continuous solution for $N \leqslant 110$. In all cases LS2 is more accurate than LS1, especially for large N values. In these experiments, LS1 and LS2 predict the fine grid solution with an error less than the fine grid approximation of the exact solution for N as large as 150: we gain therefore more than one order of convergence.

This turning point problem confirms the potential of our method but is relatively easy to solve. Let us consider now the lid-driven cavity flow problem considered previously in Section 2.2.

4.2.2. The square cavity flow problem

We consider the $C_{\omega-\psi}$ code based on FD and test case described in Section 2.2. In Figs. 10–12, we report L_∞ and L_2 errors for stream and vorticity functions with Reynolds number $Re = 20$, 100, 400. In Figs. 10–12, we take as the exact solution to measure the error, the direct numerical solution on grid 101×101 for $Re = 20$, and those on 111×111 grid for $Re = 100$ and $Re = 400$. The goal is to approximate accurately a grid solution on G4 finer than G3. As before, we denote $N \times N$ the size of G4. For $Re = 20$, the grids G1, G2, G3 are, respectively, $11 \times 11, 21 \times 21$, 31×31. For larger Reynolds number, we have, for obvious reasons, to increase the coarse grid dimension. The three coarse grids are $21 \times 21, 31 \times 31, 41 \times 41$ for $Re = 100$ and 41×41, 51×51, 61×61 for $Re = 400$. Since ω can be large and Ψ is small, we give, in fact, relative errors instead of absolute values. We recall also that Figs. 1–3 report on the order of convergence for the same data and parameters. The first two lower Reynolds number computations have been done with central differences. The $Re = 400$ case has been performed with first-order upwinding.

For the parameter of the experiment considered here, we look at data that have unsatisfactory accuracy. The RE, R2 improves the G3 grid solution in all cases except for small N. More puzzling is the fact that R2 gives better results as N increases, with fixed G2 and G3 data. This is an artifact that can be explained as follows. First we consider the exact solution to be the same as the direct numerical solution on a fine grid, and second the order of convergence is far from the theoretical asymptotic estimate (see Figs. 1–3). For larger N values the R2 error will eventually increase. We report here only on LS2 results. Further, we use 3 Newton loops in each case, and did check that more Newton loop result in little improvements.

Our first experiments with LS1 and LS2 based on the method described so far do not improve the RE method but rather give the same level of error. The result presented here needs a tricky additional post-processing. We found it critical to relax the interpolated solutions G1, G2 and G3 on G4 via the spline method, with few steps of the time integration of the Navier–Stokes equation. In all cases then LS2 gives significantly better results than R2 for the vorticity function. This relaxation procedure act as a smoother, and was not necessary for the previous turning point problem. We did use 10 times steps (respt. 20 and 40) for $Re = 20$ (respt. $Re = 100$ and $Re = 400$). The artificial time step is given by the CFL condition on the convective terms. We speculate that ω–ψ problem requires smoother interpolated data than spline interpolation can bring, because, it is in fact a fourth order problem on the stream function. In particular, we did check that the few relaxations steps that we post-process on spline interpolated solution from G1, G2, G3 data had very little effect on the size of the residual. Further, we verified that doubling the number of relaxation did not change the final result.

LS2 improves also the stream function prediction. However this is not necessarily true in our last example with $Re = 400$, apparently due to the fact that the accuracy of G1, G2, G3 data is poor. As a matter of fact R2 should give in theory worse results than R1. But this is not the case in the results of Fig. 12. Since we use first-order upwinding in this test case, it shows that the error is dominated by the viscous effect.

A refined approximation of the weight function α and β in this region may lead to significant improvement of the method. We have now a basic set of tools as comparisons of RE or Least square approximations that can be combined to control the accuracy and select the best data. This should be a topic of further investigations.

5. Conclusions

We have presented a new extrapolation method for PDEs that is more robust and accurate than RE applied to numerical solutions of PDE problems with inexact or varying convergence order. This generalized extrapolation should also be a better tool for code verification than RE when the convergence order of a CFD code is space dependent. There are still many open questions brought by this paper. In particular a posteriori estimates and adaptive domain decomposition may lead to better choices of representation of the unknown weight function in the extrapolation formula. Criteria to relax the constraint on the accuracy of the coarse grid data for efficient least square extrapolation need be further developed. In future work, we plan to extend the application of our least square extrapolation method to more complex flow problems, especially with general geometry via fictitious domain technique. For large problem, it should also be interesting to exploit the potential parallelism of our post-processing algorithm and accelerate cascade algorithm with our extrapolation technique.

References

[1] N. Bourbaki, Fonctions d'une variable reelle, 336p, reprinted, Eds Masson–Dunod, Paris, 1976.

[2] M. Boursier, C. Francois, Compte rendu Recherche Aerospatiale (131) (1969) 23–33.

[3] M. Casey, T. Wintergerste (Eds.), Ercoftac Special Interest Group on Quality and Trust in Industrial CFD – Best Practice Guidelines, ver. 1.0, Published by Ercoftac, January 2000.

[4] S.E. McCormik, Multilevel Adaptive Methods for PDES, SIAM, Philadelphia, 1989.

[5] A. Ecer, M. Garbey, M. Hervin, On the design of robust and efficient algorithms that combine Schwarz method and multilevel grids, in: C.B. Jenssen et al., 12th Int. Conf. Parallel CFD2000, Elsevier, 2000, pp. 165–172.

[6] M. Garbey, Some remarks on multilevel method, extrapolation and code verification, in: Proceedings of 13th International Conference on Domain Decomposition, Lyon Oct. 2000 to be published by CIMNE.

[7] M. Garbey, On some applications of the superposition principle with Fourier basis, SIAM Sci. Comput. 22 (3) (2000) 1087–1116.

[8] R. Glowinski, T.W. Pan, J. Periaux, Fictitious domain methods for incompressible viscous flow around moving rigid bodies, in: J.R. Whiteman (Ed.), The Mathematics of Finite Elements and Applications, Highlight, Wiley, Chichester, England, 1996, pp. 155–174, 96.

[9] G. Golub, C.F. Van Loan, Matrix Computations, second ed., Johns Hopkins University Press, Baltimore, MD, 1989.

[10] D. Gottlieb, C.W. Shu, On the Gibbs phenomenon and its resolution, SIAM Rev. 39 (4) (1997) 644–668.

[11] P. Henrici, Elements of Numerical Analysis, Inc., New York, 1964.

[12] C. Hirsch, Numerical Computation of Internal and External Flows, Vol. II: Computational Method for Inviscid and Viscous Flows, New York, 1990.

[13] A.G. Hutton, M.V. Casey, Quality and trust in industrial CFD – a European initiative, in: 39th AIAA Aerospace Sciences Meeting, 8–11 January, 2001/Reno, NV, AIAA Paper 2001-0656.

[14] M. Israeli, L. Vozovoi, A. Averbuch, Spectral multidomain technique with local Fourier basis, SIAM Sci. Comput. 8 (1993) 135–149.

[15] D. Leservoisier, G. Paul Louis, A. Dervieux, Métrique continue et Optimisation de Mailliage, Inria report RR 4172, (2001), 21 pp. Available from <http://www.inria.fr>.

[16] Y. Kuznetsov, G. Marchuk, Methodes iteratives et fonctionelles qadratiques, in: J.L. Lions, G.I. Marchuk (Eds.), Méthodes de L'Informatique-4, Dunod, Paris, 1974, pp. 3–132.

[17] W.L. Oberkampf, F.G. Blottner, D. Aeshliman, Methodology for computational fluid dynamics code verification and validation, in: 26th AIAA Fluid Dynamic Conference, June 19–22, 1995/San Diego, CA, AIAA Paper 95-2226.

[18] R. Peyret, T.T. Taylor, Computational methods for fluid flow, Springer Series in Computational Physics, 1985.

[19] P.J. Roache, Verification and Validation in Computational Science and Engineering, Hermosa Publishers, Albuquerque, New Mexico, 1998.
[20] D. Braess, Finite Elements: Theory, Fast Solvers and Application in Solid Mechanics, Cambridge University Press, Cambridge, 1997.
[21] W.H. Press, Al, Numerical Recipies in C, second ed., Cambridge University Press, Cambridge, 1994.
[22] D.C. Montgomery, E.A. Peck, Introduction to Linear Regression Analysis, second ed., Wiley, New York, 1992.
[23] B. Fornberg, A Practical Guide to Pseudospectral Methods, Cambridge Monographs on Applied and Computational Mathematics, 1998.
[24] Y. Saad, Iterative Methods for Sparse Linear Systems, PWS Publishing Company, 1995.
[25] W. Shyy, Computational Modeling for Fluid Flow and Interfacial Transport, Elsevier, New York, 1994.
[26] W. Shyy, S.S. Thakur, H. Ouyang, J. Liu, E. Blosch, Computational Techniques for Complex Transport Phenomena, Cambridge University Press, Cambridge, 1997.
[27] W. Shyy, M. Garbey, A. Appukuttan, J. Wu, Evaluation of Richardson extrapolation in computational fluid dynamics, Numer. Heat Transfer, Part B: Fundam. 41 (2002) 139–164.
[28] J.F. Thompson, Z.V.A. Warsi, C.W. Mastin, Numerical Grid Generation: Foundation and Applications, North-Holland, New York, 1985.
[29] J.F. Thompson, F.C. Thames, C.W. Mastin, Automatic numerical generation of body fitted curvilinear coordinate systems for fields containing any number of arbitrary two dimensional bodies, J. Comput. Phys. 15 (1974) 299–319.